基于深度学习的判决结果倾向性分析 *

王业沛 [1]，宋梦姣 [2]，王　譞 [3]，赵志宏 [4]

(南京大学 软件学院，南京 210093)

摘　要：裁判文书中判决结果的倾向性分析是指判断裁判文书中的判决结果是否支持原告的诉讼请求。判决结果的倾向性分析对于裁判文书的规范化、推荐诉讼律师等后续工作有重大的意义，但是缺少有效的分析模型。为了充分利用海量数据的裁判文书，提出了一个判决结果倾向性分析的模型。从半结构化的裁判文书中抽取出关键特征，利用模糊匹配的方式对判决结果中的多重实体进行识别和清洗，将处理结果交由基于 LSTM 的深度学习神经网络进行倾向性判断。通过对三种案由的数据集进行实验，该模型的准确率最高可达 98.3%，验证了该模型在判决结果的倾向性分析任务中具有很高的有效性。

关键词：深度学习；长短期记忆模型；判决结果；倾向性分析

中图分类号：TP391　　doi: 10.3969/j.issn.1001-3695.2017.08.0713

Orientation analysis of judgment results based on deep learning

Wang Yepei[1], Song Mengjiao[2], Wang Xuan[3], Zhao Zhihong[4]

(*Software Engineering*, *NanJing University*, *NanJing* 210093, *China*)

Abstract: The orientation analysis of judgment results in the judgment documents is analyzing whether the results support the plaintiffs' claims. The orientation analysis of judgment results is of great significance to the standardization of the judgment documents and the recommendation of litigation lawyers. However, there is no effective analysis model. In order to make full use of the massive data of the judgment documents, this paper designs a model for orientation analysis: we extracted the key features from the semi-structured judgment documents, and identified and cleaned the multi-entities in the judgment results by fuzzy matching. The results are processed by deep learning neural network based on LSTM for analyzing orientation. By experimenting with the data set of three different kinds of cases, the accuracy of the model is up to 98.3%, which verifies the validity of the model in the orientation analysis task of the judgment results.

Key Words: deep learning; LSTM; judgment results; orientation analysis

0　引言

裁判文书是法院代表国家行使审判权对具体案件的实体或程序问题作出具有法律效力的权威性书面结论。裁判文书通过书面的形式记录裁判过程和裁判依据、裁判理由和裁判结果[1]。裁判文书中判决结果的倾向性分析是指分析裁判文书中的判决结果，判断其是否支持原告的诉讼请求。裁判文书中判决结果的倾向性分析结果可用来判断裁判文书中术语使用是否合理、分析诉讼律师的胜利率，这对于裁判文书规范化、诉讼律师推荐等应用场景有着重大作用。然而，裁判文书是半结构化的文本，其内容的不规范、人工记录的误差、术语使用的不一致等问题，为判决结果倾向性分析带来了很大的挑战。

判决结果的倾向性分析类似短文本的情感分析问题。首先，两个任务的研究对象都属于短文本范畴，字数通常不超过 200 个字符。其次，情感分析主要涉及情感信息单元抽取和情感信息的倾向性分析，而判决结果倾向性分析涉及判决信息单元的抽取及其倾向性分析。因此，可以采用类似短文本情感分析的方法来解决判决结果的倾向性分析问题。

基于字典规则的方法和基于机器学习的方法是解决短文本情感分析的常用手段。由于基于字典规则的方法在不同类型或主题的语料上难以移植和泛化，同时过度依赖于专家的领域知识，近年来逐渐被基于机器学习的方法所替代或融合。而其中基于深度学习的方法不需要先验的知识，利用深度神经网络从词向量中学习出特征，并最终生成语言模型。

与短文本情感分析不同的是，判决结果本身是被包含在裁判文书这样半结构的文本中，无法直接获得，同时由于判决结

基金项目：江苏省产学研前瞻性联合研究项目（BY2015069-03）；中国博士后创业人才支持计划资助项目（BX201700121）

作者简介：王业沛（1993-），男，江苏扬州人，硕士研究生，主要研究方向为深度学习、自然语言处理（wangyepei1993@163.com）；宋梦姣（1992-），女，硕士研究生，主要研究方向为自然语言处理；王譞（1986-），男，讲师，博士，主要研究方向为深度学习、数据挖掘；赵志宏（1975-），男，教授，博导，主要研究方向为计算机网络、数据仓库与数据挖掘.

果的对象实体对倾向性结果有决定性作用，而判决结果中往往使用的是人名等称谓，而非统一的法律实体，因此需要准确地识别出判决的多重实体并进行清洗。本文为了解决上述问题，从裁判文书中抽取出关键特征，并利用模糊匹配的方式识别判决结果中的多重实体，采用基于长短期记忆模型的深度学习方法，设计了一个判决结果倾向性分析模型。仅使用少量标注的判决结果作为模型的训练集和测试集，通过实验对比基于词典的规则方法、基于深度学习的方法在判决结果倾向性分析中的准确率，验证其有效性。

本文的贡献主要包括：a)将深度学习应用到判决结果的倾向性分析中，并验证其有效性；b)在判决结果倾向性分析任务中，探究神经网络的深度对分类结果准确度的影响;c)利用模糊匹配，解决判决结果中多重实体的识别问题;d)针对裁判文书的语言特点，设计了一个包括文本预处理在内的实验模型，训练完成后的最终模型只需要输入裁判文书，即可得到结果标签，中间步骤无须人工参与。

1　情感分析的相关研究

近年来，情感分析，又称极性分析，已经逐渐发展成为 NLP 领域中最热门的研究问题之一。研究方法也从一开始的基于词典的规则方法，逐步转变成基于机器学习的方法。

基于词典的规则方法，通常需要先构建情感词典，再根据测试文本中的情感词在词典中的先验情感进行整条文本的情感计算。这种方法的问题包括：a)词典无法覆盖所有的情感词汇，特别是快速更新的网络词汇;b)情感词汇本身在不同的语境下可能有多重含义。针对第一个问题，Turney 等人[2]提出了根据待测文本中的词汇与种子词典中词汇的关联度，来判断整条文本的情感极性；Mohammad 等人[3]尝试生成适合社交媒体情感分析的词典。针对第二个问题，Jijkoun 等人[4]提出生成面向主题的情感词典，从而减少因主题的发散，带来的语义多样的情况。

基于机器学习的方法在 2002 年，就由 Pang[5]首次应用到情感分析领域。机器学习方法通常是将情感分析转换成模式分类的问题，建立分类模型，对情感极性作出预测。在建立模型时，需要事先标注好数据。文献[6,7]利用传统的机器学习方法，包括支持向量机、朴素贝叶斯等，通过组合不同的分类器来提高准确率。文献[8,9]尝试将基于字典的规则和传统机器学习相结合，也取得不错的进展。近年来，以循环神经网络 RNN[10]为核心思想的各种深度学习模型，包括 Hochreiter 等人提出了 LSTM 模型[11]，减少运算量的 GRU 模型[12]，可以挖掘出更多上下文信息的双向 LSTM[13]，被纷纷应用到情感分析任务中，并取得了不错的效果。目前这些研究大多集中在英文文本，而中文研究主要针对微博进行情感分析。Tang 等人[14]提出使用 word embedding 来表示词信息，Vo 等人[15]使用基于分布式词表示和深度学习的特征抽取方法并取得很好的分类效果。滕飞等人[16]提出将 LSTM 模型与主题融合，梁军等人[17]提出将 LSTM 和极性转移模型组合，张冲[18]设计了 attention_based LSTM 模型。在判决结果倾向性分析任务中尚未有人使用基于机器学习的方法，包括深度学习，进行分析。

2　基于深度学习的判决结果倾向性分析模型

本文设计的基于深度学习的判决结果倾向性分析模型如图 1 所示。

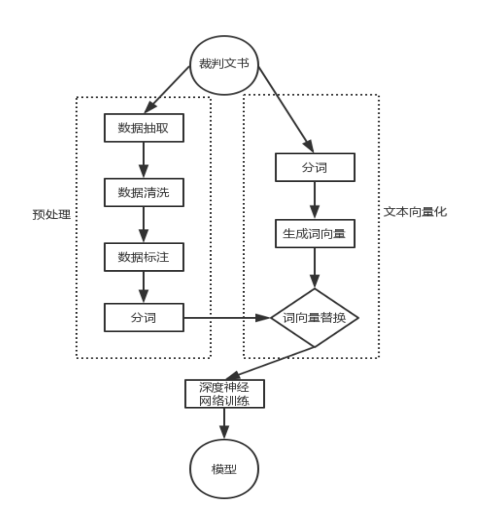


图 1　基于深度学习的判决结果倾向性分析模型

2.1　裁判文书预处理

裁判文书是半结构化文本，通常的结构如图 2 所示。

【标题】

原告：A 有限责任公司。住所地：**省**市***。

【其他内容 1】

被告：B 有限责任公司。住所地：**市***。

【其他内容 2】判决如下：

一、B 有限责任公司于本判决生效之日起十日内向 A 有限责任公司偿还借款本金 10 亿元及相应利息（自 2013 年 12 月 18 日起至本判决确定的给付之日止以上述本金为基数，按年利率 24%计付，B 公司已付利息 139638888.89 元从中扣减）;

二、驳回 A 公司的其他诉讼请求。

【其他内容 3】

图 2　裁判文书示例

裁判文书中包含了很多与倾向性分析无关的内容，如图 2 中【其他内容 1】、【其他内容 2】包含了裁判过程、裁判理由等信息，【其他内容 3】包含了裁判依据等信息。同时由于裁判结果中频繁地使用人名、公司名等称谓，而非“原告”、“被告”这样的法律词汇，在后续的分词阶段中，过长的公司名、机关名会被拆分成多个词语，导致信息的丢失，并最终影响深度训练的效果。因此，在分词前，从裁判文书中抽取出原告、被告的名称，并将判决结果中的名称替换成法律用词。

裁判文书的预处理主要分为以下几个步骤：

a)**数据抽取**。从裁判文书中抽取出原告、被告、判决结果等

关键特征。由于裁判文书半结构化的特点，抽取出关键特征所在的段落比较容易，从段落中抽取出准确的特征则需要根据特征上下文设计不同的正则匹配条件。

b)**数据清洗**。利用模糊匹配的方式，识别判决结果中的人名、公司名等称谓，并使用相对应的“原告”、“被告”等法律用语代替。

在这个步骤中，有一些判决结果中的公司名与原告、被告中的名称不完全一致，比如抽取出的原告名称为“北京**工程技术有限公司”，而在判决结果中使用“**工程技术有限公司”的称谓，这些称谓通常是全称的子串，因此在最终实验的数据清洗过程中使用的是最长公共子串算法进行模糊匹配。W 是所有原告与被告名称的集合，s_k是第 k 个名称w_k与判决结果的最长公共子字符串，r_k是第 k 个最长公共子串s_k与第 k 个名称w_k的长度比。

$$W = \{w_1, w_2, \cdots, w_n\}$$

$$r_k = \frac{length(s_k)}{length(w_k)}$$

选择集合$\{r_k\}$中的最大值所对应的身份，“原告”或则“被告”，替换判决结果中的最长公共子串。

c)**数据标注**。将上一步骤得到的判决结果进行人工标注，分别为“支持原告”和“不支持原告”。针对一些特殊情况，本文制定了额外的规则，如表 1 所示。

表 1　特殊情况标注规则

规则	标注
单条判决结果中，部分支持原告，则判定支持原告。 【例句】被告于本判决生效之日起十日内退还原告工程保证金 1376505 元，驳回原告其余诉讼。	支持原告
撤诉判定为支持原告。 【例句 1】准许原告撤诉。 【例句 2】本案按原告撤回起诉处理。	支持原告
驳回被告的反诉请求判定为支持原告，同理，驳回原告的反诉请求为支持被告。 【例句】被告对原告提出的反诉不予受理。	支持原告

标注时，由 3 人进行人工标注，每条判决结果的标注由 3 人标注的结果综合判定，从而降低人工出错的可能性。

d)**分词**。将完成上述步骤的判决结果进行分词，作为下一阶段的输入。

2.2　文本向量化

文本向量化是将分词结果使用数值向量表示。文本向量化一般有两种方式：One-hot representation 和 Distributed representation。本文采用了 Distributed representation，因为这种表示方法使用低维度向量表示每个词，并且由于将词义分散到不同维度上，使得可以根据向量判断词语之间的相似度。

目前已经有一些开源的词向量了，但是由于训练的语料场景与本文需要的裁判文书场景相去甚远，同时，为了充分利用裁判文书的内容，本文重新生成了一份词向量，用来实现文本向量化。步骤如下：

a)对裁判文书进行分词；

b)训练得到词向量；

c)将上一阶段的分词结果使用词向量表示；

2.3　深度神经网络

图 3 是本文设计的深度神经网络模型，将词向量表示的分词结果作为 LSTM 网络的输入。由于倾向性分析的最终输出是分类标签，因此只需要考虑 LSTM 最后一个单元的输出结果。又因为输出结果是一个向量，所以额外增加了一层隐藏层对向量进行特征选择，最终输出的标签使用 sigmoid 激活函数获得。整个深度神经网络的训练完成后，会得到最终的模型。本文分别设计了单层 LSTM、两层 LSTM、三层 LSTM 来分析神经网络的深度对于判决结果倾向性分析结果的影响，图 3 中虚线标注的 LSTM 层表示在不同实验中 LSTM 层数会进行增减。

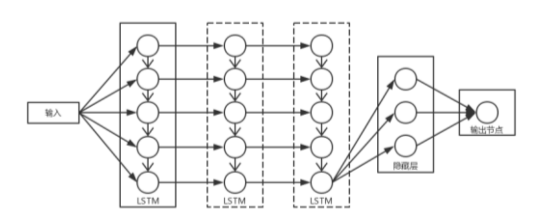


图 3　深度神经网络模型

本文使用了 Hochreiter 等人提出了 LSTM 模型作为核心，是因为对于 RNN 而言，Bengio 等人[19]发现 RNN 会出现梯度消失的问题，导致后面时间节点对前面时间节点的感知力下降。

LSTM 在 RNN 的基础上新增了细胞状态和门两个概念。细胞状态会在整个 LSTM 隐藏层中传输，存储在其中的信息不会丢失，但可以通过不同的门对其中的信息进行增删。门结构是用来选择信息的结构，LSTM 中包含了 3 种门，分别是遗忘门、输入门、输出门。如图 4 所示。

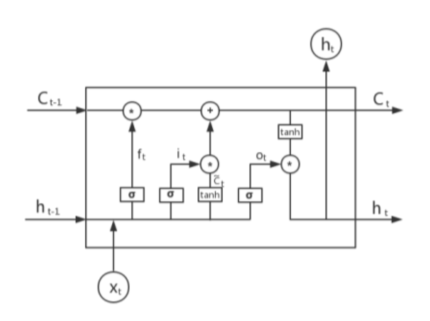


图 4　LSTM 节点内部结构

图 4 展示了 LSTM 节点的内部结构。x_t，h_t，$\tilde{C}_t$，C_t是第 t 时刻的输入，输出，候选细胞状态和细胞状态，f_t，i_t，o_t是第 t 时刻的遗忘门、输入门、输出门的结果。它们的计算公式如下：

遗忘门：$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f)$

输入门：$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i)$

输出门：$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o)$

状态候选值：$\tilde{C}_t = tanh(W_c x_t + U_c h_{t-1} + b_c)$

细胞状态更新：$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$

$$h_t = o_t * tanh(C_t)$$

其中：W_f，W_i，W_c，W_o，U_f，U_i，U_c，U_o是权重矩阵，b_f，b_i，b_c，　b_o是偏移向量，σ是 sigmoid 函数。

遗忘门用来控制从细胞状态中丢弃的内容，诸如判决结果中赔偿金额、赔偿时间，这些信息对标签的最终判定是没有影响的，因此在训练的过程中，类似信息会被遗忘。输入门用来决定哪些新信息需要增加到细胞状态中，例如“驳回【空格】被告【空格】反诉【空格】请求”，训练时，“被告”一词是“驳回”的对象，对最终的标签是有决定性影响的，因此会被更新到细胞状态中。输出门根据当前时刻的输入和细胞状态控制当前时刻的输出内容。

3　实验与结果分析

3.1　语料数据及评价标准

在判决结果倾向性分析的任务中，目前尚未有标准的测试语料库，因此本文所使用的语料数据是从中国裁判文书网（http://wenshu.court.gov.cn/）上下载的裁判文书。由于民事案件的判决结果通常不是“一边倒”的情况，即判决结果中有支持原告诉讼请求的内容，也会有不支持原告的内容，因此本文选择了民事案件中占比最多的三种案由：合同纠纷、侵权责任纠纷、婚姻家庭纠纷，每种案由下载了 500 份裁判文书。裁判文书预处理之后得到的最终标注数据集，统计如表 2 所示。

表 2　标注数据集统计

	支持原告	不支持原告	合计
合同纠纷	990 (81.95%)	218 (18.05%)	1208
侵权责任纠纷	674 (71.17%)	273 (28.83%)	947
婚姻家庭纠纷	478 (80.57%)	116 (19.43%)	594
合计	2142 (77.92%)	607 (22.08%)	2749

本文最终的实验语料是带标签 2749 条裁判文书的判决结果。

本文采用了准确率 A（Accuracy）作为评测指标，其中n_{right}，n_{wrong}分别是预测正确和错误的数量。

$$A = \frac{n_{right}}{n_{right} + n_{wrong}}$$

3.2　实验设计及结果分析

本次实验设计了四组实验，分别是基于深度学习的方法，包括基于单层 LSTM，两层 LSTM，三层 LSTM，和基于词典的规则方法（对比实验）。

在判决结果中，由于判决的对象无法固定，如“驳回原告”或者“驳回被告”，这两种表达需要根据句中“驳回”的具体对象才能判断倾向性，因此与情感分析中只考虑情感词汇不同，本文的对比实验中使用了由词语搭配构成的词典。

通过不断调整参数训练深度学习模型，直至在测试集上取得较好的实验结果，对比基于词典方法的实验结果，统计如表 3，LSTM1,LSTM2,LSTM3 分别表示单层、两层、三层 LSTM 结构，P、N 表示“支持原告”、“不支持原告”这两种分类标签。可以发现，基于深度学习的方法的精确度总是优于基于词典的方法，标签为 P 的精确度优于标签为 N 的精确度，LSTM 层数的增加会提升精确度。

表 3　深度学习方法和词典方法的实验结果

	P	**N**	**合计**
基于词典	0.93	0.904	0.924
LSTM1	0.984	0.941	0.976
LSTM2	0.984	0.950	0.978
LSTM3	0.989	0.96	0.983

图 5 展示了神经网络层数与单个迭代训练时间的关系。单个训练时间随着 LSTM 层数的增加呈线性递增。

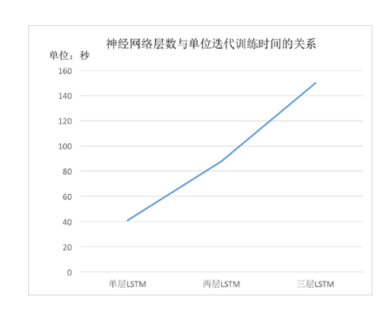


图 5　神经网络层数与单位迭代训练时间的关系图

表 4 展示了在不同数据集下，单层 LSTM 与词典方法的实验结果。其中数据集 C，TL，WF 表示合同纠纷、侵权责任纠纷、婚姻家庭纠纷这三类判决结果集，

表 4　单层 LSTM 和词典方法在不同数据集下的实验结果

数据集	基于深度学习			基于词典		
	P	N	合计	P	N	合计
C	0.982	0.978	0.981	0.9	0.936	0.907
TL	0.971	0.932	0.959	0.95	0.894	0.934
WF	0.987	0.889	0.968	0.941	0.862	0.926
全体	0.984	0.941	0.976	0.93	0.904	0.924

由实验结果可以发现：

a)针对不同民事案由的裁判文书，在正向和负向两个分类结果中，基于深度学习的方法总是优于基于词典的方法，并且全局准确率最高可达到 98.3%。证明本文设计的基于深度学习的方法在判决结果倾向性分析任务中有很好的可扩展性。

b)神经网络层数的增加可以提高倾向性分析的准确度，但是单个迭代训练的时间随网络层数线性增长。虽然基于三层 LSTM 的训练模型比基于单层 LSTM 的模型在准确率上提高了

0.7%，但是训练时间却增加了两倍。因此，在判决结果倾向性分析任务中，基于单层 LSTM 的深度学习神经网络的综合性能最好。

c)标签为“不支持原告”判决结果的准确率普遍低于“支持原告”的准确率，这主要是因为“不支持原告”的这类语料过少，深度学习模型未能充分学习预料中各种表达方式，导致预测时准确率较低。

d)婚姻家庭纠纷中，“不支持原告”这类判决结果的准确率最低，主要原因是仅仅根据判决结果的语义，是无法准确预测此类数据的标签。如判决结果为“判决被告承担子女抚养权”，这类判决结果需要根据原告的具体诉讼请求是希望被告承担抚养权，还是原告自身承担抚养权，才能判定这种判决结果的分类是“支持原告”还是“不支持原告”。

4 结束语

本文首次将深度学习引入对判决结果的倾向性分析任务中，尝试从非结构化的文本中提取关键特征，利用模糊匹配方法解决判决结果中的多实体识别的问题，再通过基于多层 LSTM 的深度神经网络进行倾向性判断，将整个过程构建成一个针对判决结果的倾向性分析模型。将该模型在三种不同案由的裁判文书数据集上进行实验，都达到很高的准确率，并且优于传统的基于词典的规则方法，验证了该模型在判决结果倾向性分析任务中具有良好的可扩展性和应用价值。这对于今后的裁判文书规范化、推荐诉讼律师等工作具有重要的意义。

本文提出的模型目前适用的范围局限在判决结果倾向性分析任务中，而一般的短文本情感分类中，单条文本可能包含多种观点或情感，本模型是否适用有待进一步的探究。另一方面，在实验过程中也发现了该模型存在的一些问题。模糊匹配的算法需要优化，部分判决结果需要更多的特征才能判断出倾向性（比如婚姻家庭纠纷中子女抚养权的判决，这种情况需要将原告的诉讼请求作为分析倾向性的一个特征）。因此，下一步的工作就是设计更加完善的模糊匹配的算法以及尝试从裁判文书中抽取更多的特征作为模型的输入，从而提高模型的准确度和稳定性。

参考文献：

[1] 刘雪华. 论裁判文书公开 [D]. 重庆: 西南政法大学, 2013.

[2] Turney P D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews [C]// Proc of Meeting on Association for Computational Linguistics. [S. l.] : Association for Computational Linguistics, 2002: 417-424.

[3] Mohammad S M, Kiritchenko S, Zhu X. NRC-Canada: building the state-of-the-art in sentiment analysis of Tweets [J]. Computer Science, 2013.

[4] Jijkoun V, Rijke M D, Weerkamp W. Generating focused topic-specific sentiment lexicons [C]// Proc of Meeting of the Association for Computational Linguistics. [S. l.] : Association for Computational Linguistics, 2010: 585-594.

[5] Pang B, Lee L, Vaithyanathan S. Thumbs up?: sentiment classification using machine learning techniques [C]// Proc of Conference on Empirical Methods in Natural Language Processing. [S. l.] : Association for Computational Linguistics, 2002: 79-86.

[6] Wang S, Manning C D. Baselines and bigrams: simple, good sentiment and topic classification [C]// Proc of Meeting of the Association for Computational Linguistics: Short Papers. [S. l.] : Association for Computational Linguistics, 2012: 90-94.

[7] Li S, Huang L, Wang J, et al. Semi-Stacking for Semi-supervised Sentiment Classification [C]// Proc of Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing. 2015: 27-31.

[8] 孙建旺, 吕学强, 张雷瀚. 基于词典与机器学习的中文微博情感分析研究 [J]. 计算机应用与软件, 2014, 31 (7): 177-181.

[9] 姜杰, 夏睿. 机器学习与语义规则融合的微博情感分类方法 [J]. 北京大学学报: 自然科学版, 2017, 53 (2): 247-254.

[10] Graves A. Generating Sequences With Recurrent Neural Networks [J]. Computer Science, 2014.

[11] Hochreiter S, Schmidhuber J. Long short-term memory. [J]. Neural Computation, 1997, 9 (8): 1735.

[12] Cho K, Van Merrienboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation [J]. Computer Science, 2014.

[13] Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF Models for Sequence Tagging [J]. Computer Science, 2015.

[14] Tang D, Wei F, Yang N, et al. Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification [C]// Proc of Meeting of the Association for Computational Linguistics. [S. l.] : Association for Computational Linguistics, 2014: 1555-1565.

[15] Vo D T, Zhang Y. Target-dependent twitter sentiment classification with rich automatic features [C]// Proc of International Conference on Artificial Intelligence. [S. l.] : AAAI Press, 2015: 1347-1353.

[16] 滕飞, 郑超美, 李文, 等. 基于长短期记忆多维主题情感倾向性分析模型 [J]. 计算机应用, 2016, 36 (8): 2252-2256.

[17] 梁军, 柴玉梅, 原慧斌, 等. 基于极性转移和 LSTM 递归网络的情感分析 [J]. 中文信息学报, 2015, 29 (5): 152-159.

[18] 张冲. 基于 Attention-Based LSTM 模型的文本分类技术的研究 [D]. 南京: 南京大学, 2016.

[19] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult [J]. IEEE Trans on Neural Networks, 2002, 5 (2): 157-166.